ROLL NO: 210701268

Exp8:

Implement SVM/Decision tree classification techniques

Aim: To implement SVM/Decision Tree classification techniques in RStudio using R language.

a) SVM IN R

```
# Install and load the e1071 package (if not already installed)
install.packages("e1071") library(e1071)
# Load the iris dataset data(iris)
# Inspect the first few rows of the dataset head(iris)
# Split the data into training (70%) and testing (30%) sets set.seed(123)
# For reproducibility
sample indices <- sample(1:nrow(iris), 0.7 * nrow(iris)) train_data
<- iris[sample indices, ]
test data <- iris[-sample indices, ]
# Fit the SVM model svm model <- svm(Species ~ ., data =
train data, kernel = "radial")
# Print the summary of the model
summary(svm_model)
# Predict the test set predictions <- predict(svm model,
newdata = test_data)
# Evaluate the model's performance
confusion matrix <- table(Predicted = predictions, Actual = test_data$Species)
print(confusion matrix)
# Calculate accuracy accuracy <- sum(diag(confusion matrix)) /
sum(confusion matrix) cat("Accuracy:", accuracy * 100, "%\n")
```

```
7 sample_indices <- sample(1:nrow(iris), 0.7 * nrow(iris))
8 train_data <- iris[sample_indices, ]
9 test_data <- iris[-sample_indices, ]
10 # Fit the SVM model
11 svm_model <- svm(Species ~ ., data = train_data, kernel = "radial")
                                                                                                                               ▲ R ▼ Global Environment ▼
                                                                                                                                    Data
                                                                                                                                                                    7 obs. of 2 variables
                                                                                                                                    0 data
                                                                                                                                                                   150 obs. of 5 variables
                                                                                                                                    0 iris
                                                                                                                                                                    List of
                                                                                                                                                                                12
   13 summarv(svm_model)
                                                                                                                                    Ologistic_model List of 30
   # Predictions <- predict(svm_model, newdata = test_data)

16 # Evaluate the model's performance
17 confusion_matrix <- table(Predicted = predictions, Actual = test_data$Specie

18 print(confusion_matrix)

19 mtcars

0 mtcars

0 svm_model

0 test_data
                                                                                                                                                                    32 obs. of 11 variables
                                                                                                                                    O svm_model
                                                                                                                                                                   List of 31
                                                                                                                                                                   45 obs. of 5 variables
  # Calculate accuracy
accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
cat("Accuracy:", accuracy * 100, "%\n")
cat("Accuracy:", accuracy * 100, "%\n")
                                                                                                                                    ① train_data
                                                                                                                                                                   105 obs. of 5 variables
                                                                                                                                   Values
                                                                                                                                       accuracy 0.97777777778 confusion_matrix 'table' int [1:3, 1:3] 14 0 0 0 17 1 0 0 13
                                                                                                                                                                   num [1:7] 150 160 165 170 175 180 185
Named num [1:32] 0.461 0.461 0.598 0.492 0.297...
                                                                                                                                       heights
  22:1
                                                                                                                                       predicted_probs
                                                                                                                                       predictions Factor w/ 3 levels "setosa", "versicolor",..: 1... sample_indices int [1:105] 14 50 118 43 150 148 90 91 143 92 ...
 Console Terminal × Background Jobs ×
R 4.4.1 · ~/ €
                                                                                                                                       weights
                                                                                                                                                                   num [1:7] 55 60 62 68 70 75 80
setosa 14
versicolor 0
virginica 0
Accuracy: 97.77778 %
```

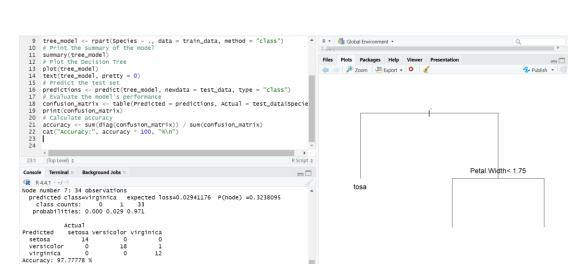
b) Decision tree in R

```
# Install and load the rpart package (if not already installed)
install.packages("rpart") library(rpart)
# Load the iris dataset data(iris)
# Split the data into training (70%) and testing (30%) sets set.seed(123)
# For reproducibility
sample indices <- sample(1:nrow(iris), 0.7 * nrow(iris)) train data
<- iris[sample indices, ]
test_data <- iris[-sample_indices, ]
# Fit the Decision Tree model tree model <- rpart(Species ~ ., data
= train data, method = "class")
# Print the summary of the model
summary(tree model)
# Plot the Decision Tree
plot(tree model) text(tree model,
pretty = 0
# Predict the test set predictions <- predict(tree model, newdata =
test data, type = "class")
# Evaluate the model's performance
```

confusion_matrix <- table(Predicted = predictions, Actual = test_data\$Species)
print(confusion_matrix)</pre>

Calculate accuracy

accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix) cat("Accuracy:", accuracy * 100, "%\n")



Result: Thus SVM and Decision Tree techniques are implemented in RStudio using R language.